Please Enter the Following Amendments:

In the Specification:

1. Please amend page 6 lines 1-2 as follows:

C1

~~Figure 27 is an inheritance diagram~~ Figures 27A, 27B, and 27C are inheritance diagrams illustrating metadata employed according to the preferred embodiment;

2. Please amend the sentence bridging lines 22-25 of page 6 as follows:

C2

The user interface software and script generation software 115 provides the screen displays and interface capability, as well as the capability to translate certain screen selections into script which is recognized by an agent/messenger code module 117, also referred to as the "agent/messenger" for brevity.

3. Please amend the sentence bridging lines 5-7 of page 7 as follows:

C3

According to the preferred embodiment, a software component including the graphical user interface to be displayed and the ~~messenger/agent~~ agent/messenger code module 117 is transmitted to the special device 113.

4. Please amend the sentence bridging lines 7-9 of page 9 as follows:

C4

It will be noted that the screen display of Fig. 5 includes a number of user-selectable screen areas, or "regions", that correspond to methods such as "sort," "search" and "compute," which may be invoked on the data.

5. Please amend the last two paragraphs of page 18 as follows:

~~Figure 27 is an inheritance diagram~~ Figures 27A, 27B, and 27C are inheritance diagrams further illustrating organization of the metadata according to the embodiment under discussion. The box labeled "UREP Named Version Object" 201 of Figure 27A represents the highest level of abstraction in the UREP and comprises a collection of data objects. The diagram of Figure ~~27~~ 27A further illustrates the basic concept that each data object contains embedded data and methods (operations) applied against the data where the data further consists of attributes and types.

C5

Figure ~~27~~ 27A illustrates a second level of abstraction 212, which includes derived classes identified as System Node 202, System Server 203, Data Source Object 204, Field Desc 205 and System Script 206. Thus, each data object has associated therewith information as to the system node(s) where it resides, the system servers within a node which access it, its attribute as being distributed or non-distributed, the field descriptors for NT files and the methods associated with it.

6. Please amend the second, third, and fourth full paragraphs of page 19 as follows:

The System Server class 203 includes all attributes and parameters regarding each server that resides on a node, where the "server" comprises the messenger, agent and assistant agent codes, i.e., everything necessary to receive a script and to execute it. The server attribute illustrated in Figure ~~27~~ 27A is the server port, which is the address (node and port) at which incoming messages are "listened for" by the messenger of the server in question.

C6

The Data Source Object 204 comprises the names used for various objects in the script. The attribute "DSC category" indicates whether the particular object is distributed (207) or non-distributed (208). As shown in Figure 27B, a~~A~~ distributed object 207 further includes subclasses 209, 210 as to the type of distribution, i.e., across SMP nodes or across nodes of a cluster. The "ObjList" attribute gives a list of the databases contained within the distributed data source name. In other words, the object name is broken down into sub-names that exist on the different nodes.

C6

Non Distributed Data Sources 208 typically are either NT files 211 or a relational database object 213, which further break down into column, index, table and size schema 215, 216, 217, 218 as known to those skilled in the art. This is shown in Figure 27C.

7. Please amend the paragraph bridging pages 19-20, and the first full paragraph of page 20, as follows:

C7

Thus, a system Node contains one or more servers, each of which hosts its own set of Data Source Objects. The relationships represented in ~~Figure 27~~ Figures 27A-27C and contained in the metadata indicate what Data Source Objects are related to which servers and thus supply the information necessary to create the local data source descriptor files at run-time.

The information represented by ~~Figure 27~~ Figures 27A-27C is preferably captured at system set-up using a graphical interface under control of a system administrator with as much automation as possible in order to avoid unnecessary data entry. For example, such an interface provides automatic scanning of the rows and columns of a relational database. Once set up, the system runs applications automatically as illustrated herein.

8. Please amend the sentence bridging lines 16-23 of page 26 as follows:

C8

In step 526, the remote agent retrieves the program from the "script" portion 206 of the associated repository (Figure ~~27~~ 27A) at the particular remote node. For example, if the remote agent receives:

mpl(xyz.exe)

it then accesses a file referenced in the metadata containing the program to be executed, which may be, for example, a "C" program, such as:

C:/program file/xyz.exe

or a URL address.